Available online at www.sciencedirect.com

SciVerse ScienceDirect

ELSEVIER

Procedia Engineering 41 (2012) 798 – 804

Engineering Procedia

www.elsevier.com/locate/procedia

International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)

Real-time Hand Gestures System for Mobile Robots Control

Ahmad 'Athif Mohd Faudzi*, Muaammar Hadi Kuzman Ali, M. Asyraf Azman, Zool Hilmi Ismail

[a]*Department of Mechatronics and Robotics Engineering, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Malaysia*

Abstract

Autonomous mobile robot navigation in an indoor environment using vision sensor is receiving a considerable attention in current robotics research activities. In this paper, a robot controlled by real-time hand gesture is proposed. This system includes an image pre-processing and feature of extraction state that consists of bounding box and Center-Of-Mass based computation. Through the feature of extraction state, the object's Center-Of-Mass and bounding box attributes are extracted to be applied for gesture sign control. This system could be used in gesture recognition for robot control applications. The result shows the developed mobile robots could be controlled successfully through hand gestures that facilitate the process of human-robot interaction.

Keywords: Hand gestures; Vision-based recognition; Image processing; Human–robot interaction

1. Introduction

Mobile robots have the capability to move around in their environment and are not fixed to one physical location. [1] The movement can be achieved using legs, wheels or other different mechanism. They have the advantage of consuming less energy and move faster than other type of locomotion mechanisms. Hand gesture recognition system plays an important role in the human-robot interactions, due to the fact that hand gestures are a natural and powerful way of communication, and can be used for the remote control of robots. Two approaches are commonly used to interpret gestures for human robot interaction, gloved-based approach and vision-based method. Using gloves, it requires wearing of cumbersome contact devices and generally carrying a load of cables that connect the device to a computer [2]. On the other hand, visual servoing is more related to gesture and environment interpretations [3, 4]. For gestures recognitions using vision-based techniques, there are many approaches that concentrate on different parts of the body. One method is considering the use of pointing gestures to locate objects. This approach operates on various feature maps (intensity, edge, motion, disparity, color) that are utilized to track head and hands [5, 6]. Another way is using the shape of the hand where the index finger from two different views in order to locate exactly the pointing destination on a 2-dimensional workspace [6]. Besides that, the approach of recognizing pointing gestures by decomposing the disparity image of a standing subject into two parts: an outstretched arm and the rest of the body can be used [5]. Davis and Shah *et al.* have developed an approach based on the 2D locations of fingertips and palms [7]. Dynamic gestures have been handled using tracking framework [8]. Contour tracking using a Kalman filter and an affine contour deformation model can be also used to identify the hand contour and shape [9]. A stochastic framework for tracking curves in visual clutter using a Bayesian random sampling algorithm was developed by Michael Israd [10].

In this paper, we developed a real-time hand gesture recognition system for mobile robot control by hand gestures in front of a static camera. By posing a hand gesture in the vision range of a robot, then by extracting the information of the camera, it is used to actuate the robot motion.

1877-7058 © 2012 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.proeng.2012.07.246

2. RoboRealm

RoboRealm is powerful robotic vision software containing all the tools needed for processing video images application for use in computer vision. It is easy to use, yet powerful enough for the cases for integrating a vision system into robots. With a combination of programming logic and control interfaces, it is possible to generate a visual processing loop that instructs a robot to move according to what it sees by a simple USB webcam. RoboRealm is used mainly to translate what a camera sees into meaningful numbers in order to cause a reaction to what the image contains [11].

3. Hardware System: Design and Mechanism

3.1. Hardware Component:

Fig. 1 illustrates the overall view of the system. The system contains of a camera as the vision sensor connected to a laptop for processing image/video. The laptop is connected to *XBee* wireless transmitter module which in turn sends data to another *XBee* receiving platform which receives the data and in turn sends data to the main controller containing a PIC chip, which is interfaced to the robot motors through a motor driver circuit.

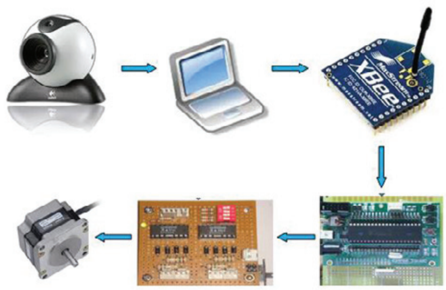

Fig. 1. System Hardware Overview

3.2. Mechanical Design

In this paper, mobile robot was designed in accordance with purposes, basic design, mechanism design based on designs, mechanism interpretation to control, and final design based on situation [12].The design of the mobile robot is simple yet convenient for the system. The mobile robot consists of three layers and the layers are placed 4 cm apart to allow for a suitable gap for circuits to be placed. Figs. 2 have shown a snapshot for the complete structure of the robot. The main board including the microcontroller and stepper motor controller circuit is placed on the bottom layer. It also holds the stepper motors on both sides of the layer and a cargo ball is placed in the front part of the layer. The second layer contains the battery, an *XBee* module circuit.

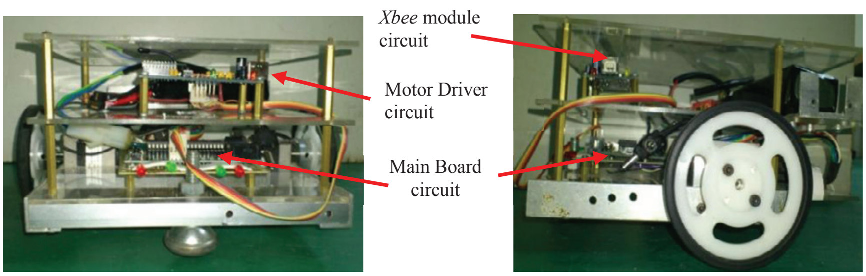

Fig. 2. (a) Front View; (b) Side View

3.3. Electronic Circuits Design

- *Main Circuit*

The schematic for main board circuit is shown in Fig. 3(a). Main board is used as the main controller for the control decision of the mobile robot. It contains PIC18F452 as the main microcontroller. PIC18F452 contains the program written in C for motors control. A snapshot of the actual circuit is as shown in Fig. 3(b).

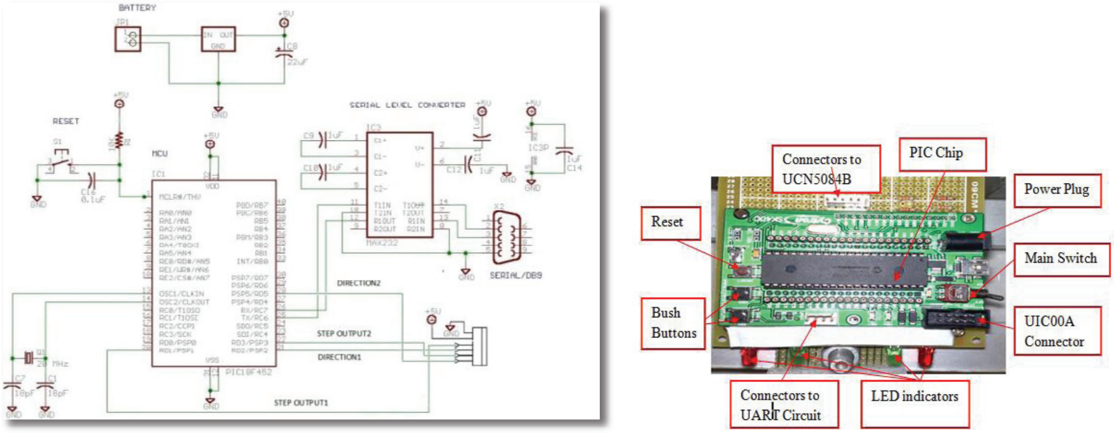

Fig. 3. (a) Main Board Schematic Circuit; (b) Main Board Circuit

- *Motor Driver Circuit*

The motor driver used for stepper motors control consists of UCN5804B as the controller to control the stepper motor speed and direction. There are three modes of operation that can be selected based on the desired operation. Those modes for operating the stepper motors are wave-drive (one-phase), two-phase, and half step. A DIP switch is used for selecting the desired mode of operation. The two-phase is normally used. The schematic for the UCN5804B circuit is shown in Fig. 4(a), while Fig. 4(b) shows the real design of the circuit. [13].

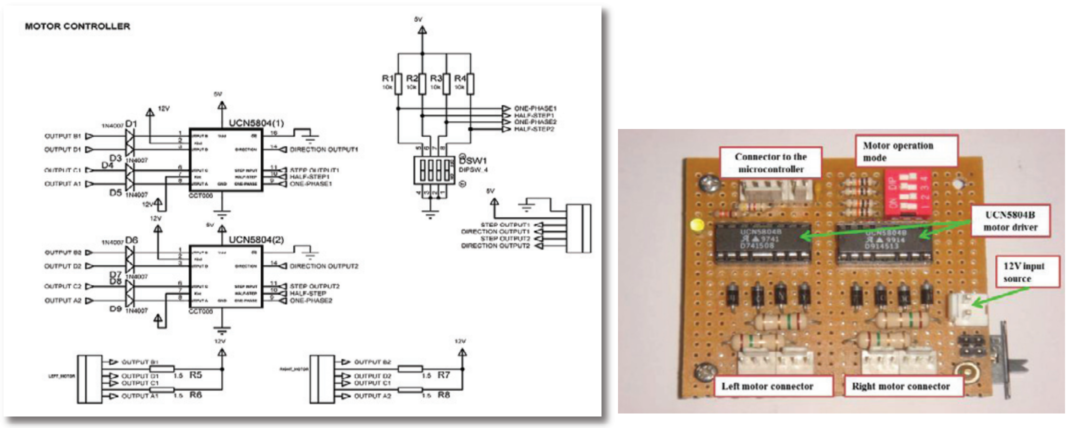

Fig. 4.(a) UCN5804B Motor Driver Schematic Circuit ; (b)UCN5804B Motor Driver Actual Circuit

- *USB to UART Converter and XBee Module*

Fig. 5(a) shows the USB UART converter circuit. This circuit performs the operation of "Sending" data through XBee in the *XBee* Transmitter Module to the *XBee* Receiver Module (Fig. 5(b)) which in turn receives the data and forward them to the microcontroller which in turn performs the action required.

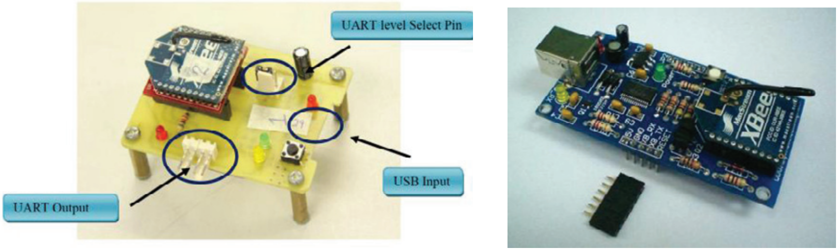

Fig. 5. (a)USB to UART Converter / *XBee* Transmitter Module; (b) *XBee* Receiver Module

4. Image Processing

Image processing using RoboRealm is utilized. The type of image processing followed is real-time processing, which has fast image processing time and is immune to the varying of the size irrespective of how far the object is from the camera. Color filtering technique is used to extract the shape of the glove. The object of interest colors to be considered is red while other colors for other objects are discarded. The object to be filtered is a red glove, hence filtering for red color objects and eliminating any other object which simplifies the process of detecting the hand gestures and postures.

4.1. Color Detection

The custom colour is predefined to be red, where only the red color channel is used and the other channels contributes for any deviation of red color. The RGB Filter filters out the red pixels from the object and eliminate for any other colors where

$$R = (R - G) + (R - B),\ G = 0, B = 0 \qquad (1)$$

Thresholding property is also available to eliminate for pixels with dark or white colour not containing enough red colour. A hysteresis level is applied to the object for eliminating any red noises produced by lighting or variations of intensity. In Fig. 6 (a) the red hand glove is segmented from the background for further processing.

4.2. Noise Filtering

A mean is used for filtering noises produced by additional factors such as lighting intensities and the presence of moving particles. The object is furthered smoothed by taking the average of pixels and becomes blurred which has the advantage of eliminating any unexpected variations of noise factors. Then a Gaussian filter is used to further smoothens the image but will preserve edges better than the more basic mean filter. The resulting object is shown in Fig. 6(b). By weighting a pixels contribution to the final pixel value this filter can better preserve edges than the mean filter which specifies equal weights to all pixels within the filter window. For a 1-D Gaussian filter the single filter values are defined as

$$G(x) = \sqrt{\frac{1}{\sqrt{2\pi}}}\, e^{-\frac{x^2}{2^2}} \qquad (2)$$

Fig. 6. (a) Red Glove filtered using RGB filter (b) Noise reduction and filtering

4.3. Blob Size Filter

The Blob Size filter is used to remove objects below a certain size. It is used to threshold the image into object/no object values

4.4. Center of Gravity

The center of gravity property is the ultimate and most important part of the image processing for determining the coordinates of the center of the object. As can be seen in Fig. 7; the hand centroid coordinates in *X* and *Y* are displayed. This can be calculated as:

$$\frac{\sum_{k=n+1}^{n} x_n}{n} \tag{3}$$

$$\frac{\sum_{k=n+1}^{n} y_n}{n} \tag{4}$$

where x_n and y_n are the coordinates of the nth pixel in the desired object and n is the total pixels in the desired object. The bounding box drawn around the hand glove is an important property that is used to eliminate red objects that are smaller than a threshold of 20 pixels which represent noise so only the desired glove color is detected.

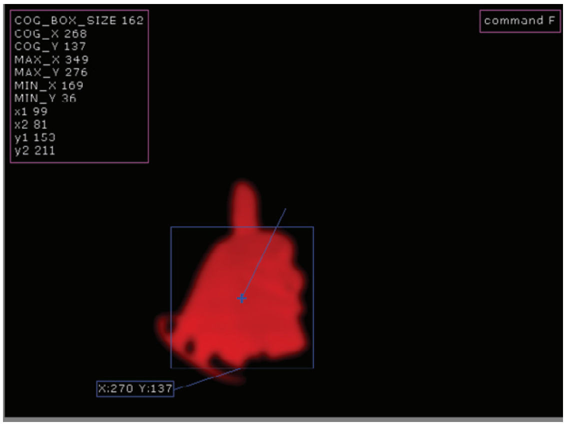

Fig. 7. Center of gravity property

5. Gestures Signs Classification

The control of motors directions and orientations is based on the principle of the longest distance of the center of the hand into the farthest point of left, right, top and bottom and comparing every one against each other, to determine which one is the longest, hence informing the motors to actuate according to the received signal. The equations which determine the distance of the center to all direction are as below:

$$xl = cogX - minx \tag{5}$$

$$yb = \frac{640(cogY - \min Y)}{480} \tag{6}$$

$$xr = maxX - cogX \tag{7}$$

$$yt = \frac{640(maxY - cogY)}{480} \tag{8}$$

where *xl* is the distance from center of max to the minimum point in the *x* coordinate, *xr* is the distance from center of max to the maximum point in the *x* coordinate, *yb*, is the distance form center of max to the minimum point in the y direction, and *yt* is the distance form center of max to the maximum point in the y direction. Since the outline of the image is 640 by 480, we have to scale the *Y* coordinate from 480 to 640 by multiplying the corresponding *y* values by 640/480.

Table 1: Control Decision Commands

Longest distance	Action	Data sent
xl	Left	L
xr	Right	R
yt	Forward	F
yb	Backward	B

Next, the input and output ports and the UART are defined and initialized. Two red LEDs will blink to indicate that the initialization has been done. It will wait for a while to check if data is ready in the UART buffer. If so then the UART receives the data and a selection loop is initiated to execute the appropriate command according to the received data. There are four possible data to be received which are as shown in Table 1 with their corresponding actions in Figs. 8(a, b) and 9(a, b). The UART is read repeatedly as long as there are signals ready in the buffer.

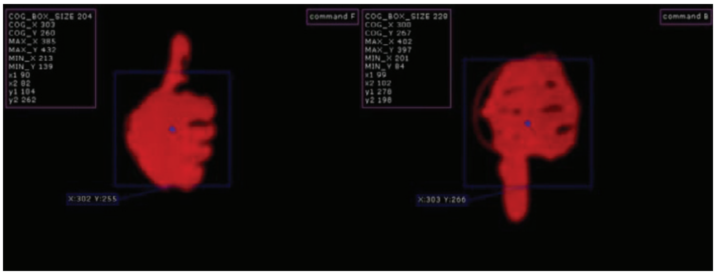

Fig. 8. (a) Forward Direction ; (b)Backward Direction

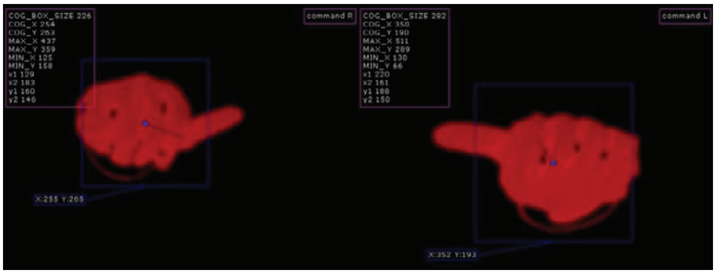

Fig. 9. (a) Right Direction; (b) Left Direction

Table 2 below depicts the various conditions of movement according to the largest distance from the center of the mass to the index finger. In row 1 it is obvious that *yb* (the distance from center of mass to the top pixels representing the index finger) is the largest among other values, hence the robot is instructed to move forward. In row 2 *Yb* is the largest hence the robot moves to the right. While in row 3 and 4 the largest are *xr* and *xl* respectively so the motors are actuated to move the robot to the right and left respectively.

Table 2: Calculation distances from center of gravity to the maximum pixels in each direction

cogX	cogY	MaxX	MaxY	MinX	MinY	xl	xr	yb	yt	Direction
386	227	506	406	275	95	111	120	176	238	F
323	290	446	448	196	58	127	123	309	210	B
381	223	592	340	211	107	170	211	154	156	R
378	228	547	334	155	112	223	169	154	141	L

6. Conclusion

Using image processing techniques for identifying hand gestures is complex. To facilitate the process of the hand gestures identification a glove-based technique through colour filtering and eliminating undesired objects was used. The hand gesture system can be used to control mobile robot using the properties of centre of gravity to identify the distance of the centre of the object to the index finger indicating which direction the robot should moves. The experimental results show that the robot is successful to move front, backward, left, right and stop based on the hand gestures input from the user.

Acknowledgment

The authors would like to give a special thanks to Universiti Teknologi Malaysia (UTM) for providing facilities required for achieving the objectives of the research, and Ministry of Higher Education (MOHE) Malaysia.

References

[1] Lee, C.H "Recurrent neuro fuzzy control design for tracking of mobile robots via hybrid algorithm", Expert Systems with Applications, 200907.
[2] Hasanuzzaman, M., V. Ampornaramveth, et al. (2004). "Real-time Vision-based Gesture Recognition for Human Robot Interaction", Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on.
[3] Sue Han Lee, S. N. C., Chee Pun Ooi, Wei Heng Siew (2011). "Real time FPGA implementation of hand gesture recognizer system." Proceedings of the 15th WSEAS international conference on Computers: Pages 217-222.
[4] Malima, A., E. Ozgur, et al. (2006). "A Fast Algorithm for Vision-Based Hand Gesture Recognition for Robot Control". Signal Processing and Communications Applications, 2006 IEEE 14th.
[5] Kai Nickel, Rainer Stiefelhagen. "Visual recognition of pointing gestures for human–robot interaction Interactive Systems Labs", Universitaet Karlsruhe, 76131 Karlsruhe, Germany. Sciencedirect journal, Volume 25, Issue 12, 3 December 2007, Pages 1875–1884.
[6] N. Jojic, B. Brumitt, B. Meyers, S. Harris, T. Huang. Detection and Estimation of Pointing Gestures in Dense Disparity Maps. IEEE Intl. Conference on Automatic Face and Gesture Recognition, Grenoble, 2000.
[7] J. Davis, M. Shah, Recognizing hand gestures, in: Proceedings of the European Conference on Computer Vision, ECCV, 1994, pp. 331–340.
[8] A. Ramamoorthy et la." Recognition of dynamic hand gestures" .Delhi, New Delhi-110016, India Volume 36, Issue 9, September 2003, Pages 2069–2081
[9] A. Blake, M. Isard, D. Reynard Learning to track the visual motion of contours Artif. Intell., 78 (1–2) (1995), pp. 179–212
[10] M.A. Isard, Visual motion analysis by probabilistic propagation of conditional density, Ph.D. Thesis, Robotics Research Group, University of Oxford, September 1998.
[11] Roborealm, robot vision system software;www.roborealm.com
[12] Do-Eun Kim, Kyung-Hun Hwang et al (2006). "Mobile Robot Design apply to Design Concept". SICE-ICASE International Joint Conference 2006 Oct. 18-2 1, 2006 in Bexco, Busan, Korea
[13] UCN5084B Motor Driver Datasheet; www.alldatasheet.com/view.jsp?Searchword=UCN5084B
[14] *Xbee* Datasheet ;www.sparkfun.com/datasheets/Wireless/Zigbee/*XBee*-Datasheet.pdf